(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（A）**　(11)特許出願公開番号

特開平10－233802

(43)公開日　平成10年(1998)９月２日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｈ０４Ｌ　12/56		Ｈ０４Ｌ　11/20	１０２Ｃ

審査請求　未請求　請求項の数20　ＯＬ　（全 11 頁）

(21)出願番号　特願平10－27551

(22)出願日　平成10年(1998)２月９日

(31)優先権主張番号　６０／０３７８４３

(32)優先日　1997年２月７日

(33)優先権主張国　米国（ＵＳ）

(31)優先権主張番号　０８／９６１１２２

(32)優先日　1997年10月30日

(33)優先権主張国　米国（ＵＳ）

(71)出願人　598077259
ルーセント　テクノロジーズ　インコーポレイテッド
Ｌｕｃｅｎｔ　Ｔｅｃｈｎｏｌｏｇｉｅｓ　Ｉｎｃ.
アメリカ合衆国　07974　ニュージャージー、マレーヒル、マウンテン　アベニュー　600－700

(72)発明者　アビジット　カマー　チョーデュリー
アメリカ合衆国，07920　ニュージャージー，スコッチ　プレインズ，ハンティントン　ロード　77

(74)代理人　弁理士　三俣　弘文

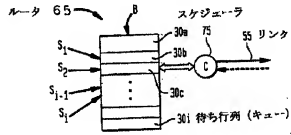
最終頁に続く

(54)【発明の名称】　ＴＣＰ接続の性能改善方法

(57)【要約】

【課題】　フィードバック制御ＴＣＰネットワークの性能を改善する。

【解決手段】　フロー毎にキューイングする装置であり、複数の待ち行列に区分けされた所定サイズのバッファと、所定速度に従って各バッファからパケットを除去し、ネットワークを介して前記パケットを伝送するスケジューラと、受信パケットを受信することができ、待ち行列が使用可能である場合、パケットを待ち行列に入力することができるバッファ内の待ち行列の使用可能性を決定するためのコントロールデバイスとからなり、コントロールデバイスは更に、最長待ち行列ファーストスキームに従って待ち行列を選択し、待ち行列が使用可能でない場合、前記受信パケットの入力を可能にするために前記選択された待ち行列からパケットをドロッピングし、これにより、リンクによる公平性を高め、パケットスループットを高める。



1

【特許請求の範囲】

【請求項１】　(i)所定サイズのバッファを複数個の待ち行列に区分けするステップと、

ここで、各待ち行列は、情報パケットを受信し、一時的に格納するための占有率b_iが割り当てられ、また、各バッファからパケットを除去し、そして、前記パケットをＴＣＰ接続を介して伝送するためのスケジューラにより処理される、

(ii)パケットが到着すると、前記パケットを受信するための待ち行列の使用可能性を決定し、前記待ち行列が使用可能である場合、前記パケットを待ち行列に入力するステップと、

(iii)前記待ち行列が使用可能でない場合、前記パケットの入力を可能にするため、待ち行列を選択し、そして、前記選択された待ち行列からパケットを解放するステップとからなり、これによりＴＣＰ接続の利用率を高めることを特徴とするＴＣＰ接続の性能改善方法。

【請求項２】　パケット到着時又は隊脱時に、各待ち行列の現在の長さq_iをトラッキングするステップを更に有する請求項１の方法。

【請求項３】　トラッキングステップは、パケットが前記待ち行列に入力されるたびに、前記待ち行列に付随するカウンタを増分するか、又は、パケットが前記待ち行列から解放されるときに、前記カウンタを減分するステップを有する請求項２の方法。

【請求項４】　前記スケジューラは所定の速度で前記各バッファからパケットを除去することができる請求項１の方法。

【請求項５】　前記バッファ待ち行列が満杯の場合、前記満杯待ち行列に宛てられたパケットが到着すると、満杯ではない１つ以上の待ち行列からバッファスペースを一時的に借りるステップを有し、前記満杯待ち行列の前記現在の長さはその名目的な占有率b_iを越える請求項２の方法。

【請求項６】　待ち行列の使用可能性を決定する前記ステップは、所定サイズの前記バッファが満杯であるか否かを決定するステップを有する請求項５の方法。

【請求項７】　前記選択された待ち行列から解放されるべき前記パケットは、前記待ち行列に内在する最も古いパケットである請求項２の方法。

【請求項８】　待ち行列を選択する前記ステップは、

(a)各待ち行列について現在の割当てb_iを確立するステップと、

(b)前記各待ち行列の現在の待ち行列長さの値q_iを取得するステップと、

(c)現在の待ち行列長さの値q_iと、各待ち行列の割当てられたバッファ占有率b_iとの間の差を計算するステップと、

(d)計算された差で最も大きな値を有する前記待ち行列を選択するステップとを有する請求項２の方法。

2

【請求項９】　待ち行列を選択する前記ステップは、

(a)各待ち行列について現在の割当てb_iを確立するステップと、

(b)現在の待ち行列長さの値q_iが各待ち行列の割当てられたバッファ占有率b_iを越える一連の待ち行列を計算するステップと、

(c)前記一連の待ち行列からランダムに待ち行列を選択するステップとを有する請求項２の方法。

【請求項１０】　前記スケジューラは異なる所定速度に従って各バッファからパケットを除去し、待ち行列を選択する前記ステップは、

(a)そのそれぞれの待ち行列長さq_iとその所定のサービス速度から各待ち行列に格納されたパケットにより経験されるキューイング遅延を計算するステップと、

(b)最も長いキューイング遅延を経験する待ち行列を選択するステップとを有する請求項２の方法。

【請求項１１】　前記選択ステップは、

(a)１つ以上のビンを確立するステップと、ここで各ビンは所定の長さの１つ以上の待ち行列に結合される、

(b)最長長さを有する１つ以上の待ち行列に結合されたビンから待ち行列をランダムに選び取るステップを更に有する請求項２の方法。

【請求項１２】　複数のソースからの情報パケットをＴＣＰ／ＩＰネットワーク内の単一の通信リンクへ通信するルータであり、

(a)複数個の待ち行列に区分けされた所定サイズのバッファと、ここで、各待ち行列は、情報パケットを受信し、一時的に格納するための占有率b_iが割り当てられる、

(b)各バッファからパケットを除去し、また、前記接続を介して前記パケットを伝送するためのスケジューラと、

(c)前記バッファの前記待ち行列が使用可能である場合、前記待ち行列に受信バッファを入力し、更に、前記バッファの前記待ち行列が使用可能でない場合、待ち行列を選択し、前記受信パケットの入力を可能にするため、前記スケジューラが前記選択された待ち行列からパケットを解放できるようにするための、前記バッファ内の待ち行列に使用可能性を決定するコントロール手段と、からなること特徴とするルータ。

【請求項１３】　パケットが入力されるたびに、又は前記待ち行列から解放されるたびに、前記待ち行列の現在の長さq_iをトラッキングするための、前記各待ち行列に結合された手段を更に有する請求項１２のルータ。

【請求項１４】　前記トラッキング手段は、前記待ち行列に結合されたカウンタデバイスを有し、前記カウンタデバイスは、パケットが前記待ち行列に入力されるたびに増分し、パケットが前記待ち行列から解放されるときは減分する請求項１３のルータ。

【請求項１５】　前記コントロール手段は、

(a)前記各待ち行列の現在の待ち行列長さの値q_iを取得する手段と、

(b)現在の待ち行列長さの値q_iと、各待ち行列について割当てられたバッファ占有率b_iとの間の差を計算する手段とを有し、計算された差で最大の値を有する前記待ち行列を選択する請求項１３のルータ。

【請求項１６】　前記コントロール手段は、

(a)現在の待ち行列長さの値q_iが各待ち行列について割当てられたバッファ占有率b_iを越える一連の待ち行列を計算する手段と、

(b)前記一連の待ち行列からランダムに待ち行列を選択する手段とを更に有する請求項１３のルータ。

【請求項１７】　前記選択された待ち行列から解放されるべき前記パケットは、前記待ち行列に内在する最も古いパケットである請求項１２のルータ。

【請求項１８】　前記スケジューラは異なる所定速度に従って各バッファからパケットを除去し、前記コントロール手段は、

(a)そのそれぞれの待ち行列長さq_iとその所定のサービス速度から各待ち行列に格納されたパケットにより経験されるキューイング遅延を計算する手段と、

(b)最も長いキューイング遅延を経験する待ち行列を選択する手段とを更に有する請求項１３のルータ。

【請求項１９】　１つ以上のソースからの情報パケットをリンクを介して宛先へ流すことができるフィードバック制御ＴＣＰ接続からトラヒックを搬送するＩＰネットワークのためのフロー毎のキューイング装置であり、

(a)複数個の待ち行列に区分けされた所定サイズのバッファと、ここで、各待ち行列は、情報パケットを受信し、一時的に格納するための占有率b_iが割当てられる、

(b)所定の速度に従って各バッファからパケットを除去し、また、前記ネットワークを介して前記パケットを伝送するためのスケジューラと、

(c)前記受信パケットを受信することができ、前記待ち行列が使用可能である場合、前記パケットを前記待ち行列に入力することができる前記バッファ内の待ち行列の使用可能性を決定するためのコントロールデバイスとからなり、前記コントロールデバイスは更に、最長待ち行列ファーストスキームに従って待ち行列を選択し、そして、前記待ち行列が使用可能でない場合、前記受信パケットの入力を可能にするために前記選択された待ち行列からパケットをドロッピングし、これにより、前記リンクによる公平性を高め、パケットスループットを高めることを特徴とするフロー毎のキューイング装置。

【請求項２０】　１つ以上のソースからの情報パケットをリンクを介して宛先へ流すことができるフィードバック制御ＴＣＰ接続からトラヒックを搬送するＩＰネットワークのためのフロー毎のキューイング方法であり、

(a)複数個の待ち行列に区分けされた所定サイズのバッファを提供するステップと、ここで、各待ち行列は、情報パケットを受信し、一時的に格納するための占有率b_iが割当てられる、

(b)所定の速度に従って各バッファからパケットを除去し、また、前記ネットワークを介して前記パケットを伝送するためのスケジューラを提供するステップと、

(c)前記受信パケットを受信することができ、前記待ち行列が使用可能である場合、前記パケットを前記待ち行列に入力することができる前記バッファ内の待ち行列の使用可能性を決定するためのコントロールデバイスを提供するステップとからなり、前記コントロールデバイスは更に、最長待ち行列ファーストスキームに従って待ち行列を選択し、そして、前記待ち行列が使用可能でない場合、前記受信パケットの入力を可能にするために前記選択された待ち行列からパケットをドロッピングし、これにより、前記リンクによる公平性を高め、パケットスループットを高めることを特徴とするフロー毎のキューイング方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明はＴＣＰプロトコルを処理する通信ネットワークに関する。更に詳細には、本発明はフィードバック制御ネットワークトラヒック用の接続ごとのキューイング(queing)及びバッファ管理スキームに関する。

【０００２】

【従来の技術】データパケットの高信頼トランスポートを実行するためのトランスポートプロトコル“ＴＣＰ”処理系は周知である。図１は、ＴＣＰ／ＩＰネットワーク（例えば、インターネット、イントラネット）における従来技術による常用のルータ装置の概要図である。

【０００３】図１において、ローカル・エリア・ネットワーク（ＬＡＮ）２０はルータ５０を介して広域ネットワーク（ＷＡＮ）６０に接続されている。一般的に、ルータ５０のアーキテクチャは、複数のデータストリーム（例えば、ＬＡＮ２０に接続されたソースデータ端末からのもの）が単一のバッファにより受信され、かつ、リンク５５を介してパケットを送信するサーバにより処理されるようなものである。

【０００４】データパケットがその宛先（例えば、広域ネットワーク６０に配置されたデータ端末）に到着すると、パケットの受信機は、受信された各データパケットについて、暗黙的に又は明快に、肯定応答“ＡＣＫ”パケットを発生する。ソース端末からのデータ送信は、先行データ送信からの受信肯定応答パケットの速度により制御される。

【０００５】従って、周知のように、ソースからのデータパケット送信の制御は、RichardW. Stevens, "TCP/IP Illustrated", Vols. 1-3, Addison Wesley 1994に概説されるように、ＴＣＰ－ＲｅｎｏとＴＣＰ－Ｔａｈｏ

ｅの、適応型スライディングウインドウスキームの２つの主要変形に従い、受信機からのデータＡＣＫパケットのフィードバックにより決定される。

【０００６】ＴＣＰは長ラウンドトリップ時間を伴う接続に対して固有の不公平性(unfairness)を示すことが確認された。すなわち、短ラウンドトリップ時間接続は、長ラウンドトリップ時間接続よりも大きな帯域幅を得てしまう。なぜなら、短ラウンドトリップ時間接続の場合のＴＣＰウインドウは、長ラウンドトリップ時間接続のウインドウよりも速く成長してしまうからである。これは、短ラウンドトリップ時間接続では接続肯定応答（ＡＣＫ）がより速く受信されるためである。この固有の不公平性を解消する効果的な解決策は未だ発見されていない。

【０００７】ＴＣＰに固有の別の問題点は、ウインドウ同期(window synchronization)現象である。例えば、２つの等しいラウンドトリップ時間接続がリンクにあるとすれば、そのリンクは飽和してしまい、バッファに蓄積が始まる。或る時点で、そのリンクに接続されたバッファはオーバーフローしてしまう。そして、これらの接続はおおよそ同時にパケット損失を経験する。結果的に、各接続はそのウインドウを事実上同時に縮小し、ネットワークスループット速度も同時に低下してしまい、その結果、ネットワークリンクは十分に活用されない。

【０００８】ＴＣＰ接続ネットワーク性能を低下させるその他の問題点は、位相効果、ボトルネックリンクによる低リバースパスの物理的な帯域幅非対称性、バーストトラヒックの生成、分離性の欠如、より侵略的なトランスポートプロトコル又は悪意的ユーザからの保護性の欠如などである。

【０００９】ネットワーク性能を改善するために実行された或る解決法は、ランダム・アーリー・ディテクション（“ＲＥＤ”）スキームである。このスキームは、バッファが飽和される前に、パケットをドロップするように機能する。バッファはモニターされ、平均バッファ占有率が特定の閾値を越えたら、パケットは、例えば、平均バッファ占有率（移動時間平均）の関数である特定の確率に従ってランダムにドロップされ、結果的に、ＴＣＰ接続はそのウインドウをランダムにカットする。

【００１０】これは、ウインドウを非同期化するのに特に有効である。特に、ＦＩＦＯ－ＲＥＤスキームでは、リンクを多数使用する接続は、一層多くのパケットをドロップし易い。従って、結果的に、その対応ウインドウサイズもカットされる。これは、このような高速度接続のバイアスを低下させる傾向がある。

【００１１】米国特許出願第０８／８５８３１０号明細書に開示された別のスキームは、リバースパスが混雑した時に、公平性とスループットを高める効果を有するＡＣＫパケットに関するフロントストラテジーからのドロップを実行する。

【００１２】非ＴＣＰ閉ループトラヒック（例えば、いわゆる“リーキー・バケット(leaky-bucket)”条件付タイプのストリーム）の場合、等しい又は異なる重みに従い、各人力接続間で帯域幅が共用されるように実行されるスケジューリングスキームに従って、接続リンクの性能を最大化するための公平待ち行列スケジューリングスキームが実行されている。

【００１３】最近の研究努力は、エンド・ツー・エンド遅延のバウンド(bound)及び分離(isolation)を保証する手段として、公平待ち行列スキームを使用することに集中的に向けられている。これに関連する研究は基本的に、非フィードバック制御リーキーバケット管理トラヒックに関するものであり、重み付公平キューイングスキームによる交換機及びルータが発達した。

【００１４】例えば、A.K. Parekh and R.G. Gallager, “A Generalized Processor Sharing Approach to Flow Control in the Single Node Case”, Proc. of INFOCOM ’92, pp. 915-924, May 1992及びD. Stiliadis and A. Varma, “Design and Analysis of Frame-based Queuing: A New Traffic Scheduling Algorithm for Packet-Switched Networks”, Proc of ACM SIGMETRICS ’96, pp. 104-115, May 1996参照。

【００１５】交換機における公平キューイング（ＦＱ）の関心は、様々なバッファ構成（例えば、物理的又は論理的に分離されたバッファ、共用バッファなどのようなバッファ構成）を処理するためのスケジューラの使用に集中している。

【００１６】

【発明が解決しようとする課題】トラヒックを制御し、フィードバック制御ＴＣＰネットワークの性能を改善するための公平キューイングスキームの実現が強く望まれている。

【００１７】更に、ＴＣＰ接続に関する相当のスループットを可能にするパケットドロップ機構を実行する公平キューイングスキームの実現が強く望まれている。

【００１８】

【課題を解決するための手段】前記課題は、本発明による、フィードバック制御ＴＣＰネットワークにおける公平キューイングスキームと共に併用されるフロー／接続毎の共用バッファ管理スキームにより解決される。

【００１９】本発明によれば、　長ラウンドトリップ時間を伴う接続に対するＴＣＰの固有の不公平性を軽減する、　異なるＴＣＰバージョンを用いる接続がボトルネックリンクを共用する場合、分離性を与える、　一層侵略的なトラヒックソース、不品行なユーザはリバースパス輻輳の場合のその他のＴＣＰ接続からの保護性を与える、　両方向トラヒックの存在下におけるＡＣＫ圧縮作用を軽減する、　ＡＣＫ損失（トラヒックのバーストを起こす）を経験するユーザに顕著な悪影響を及ぼすその他の接続から保護する、及び　全体的なリンク利用率

を低下させることなく、“欲張り”接続を伴うボトルネックを共用する双方向接続に対する低待ち時間を与える、などのようなTCPの福々の目標を達成することができる。

【0020】更に詳細には、共用バッファアーキテクチャは、各接続のための帯域幅予約保証 r_i により実行される。速度 r_i のそのバッファを完全に使用する第1の接続と、十分に利用されない（無駄いされる）、速度 r_2 の第2の接続があり、第1の接続が更に一層の帯域幅を必要とする場合、共用バッファスキームでは、r_2 接続からバッファリング（帯域幅）を借りることもできる。

【0021】第2の接続がそのバッファスペースを矯正しなければならない場合、別の利用バッファからのデータを押し出し、着信データパケットのための場所を空けなければならない。

【0022】接続毎の待ち行列（キュー）管理スキームは、共用バッファアーキテクチャにおけるパケットドロッピングメカニズム（例えば、最長待ち行列ファースト（“LQF”））をサポートし、FIFO－REDバッファ管理スキームよりも優れたTCP性能を生じる。本明細書では、公平性は、全リンク容量に対する個々のスループットの割合の平均値に対する標準偏差の比により計算する。

【0023】

【発明の実施の形態】図2は、様々なソース s_1，・・・，s_2 から発信するパケットトラヒックを処理するTCPネットワーク接続のルータ65のための接続毎の待ち行列アーキテクチャを示すブロック図である。

【0024】ネットワーク接続要素は、汎用の共用バッファメモリBを有する。このメモリBは、速度Cでリンク55上のデータパケットを処理するスケジューラ75と単一（ボトルネック）のリンク55との接続のために、“i”個の複数の待ち行列30a，・・・，iに分割されている。各バッファ接続iは、接続iの保証バッファサイズである名目バッファ割当て b_i を有する。

“フロー毎のキューイング”又は“接続毎のキューイング”として知られているアーキテクチャでは、待ち行列に入れられるべき到着パケットのきめ細かい動的分類が必要である。

【0025】“ソフト状態”アプローチは、潜在的に非常に多数のおそらく活性な接続（この場合、大多数の接続は実際には活性ではないであろう）へ導く接続状態を維持する。ネットワークノード内の関連状態はその他のガーベジ・コレクション（ごみ集め）の手段によりタイムアウト（時間切れ）を起こさせるか、矯正させるか又は遅延させるために待機される。従って、スケーラビリティ（scalability）及び共用は、接続毎のサーバーのための基本的な要件である。必要な全ての操作は、0（1）複雑さにより実行可能であり、リソース（バッファ又は帯域幅）は所定の接続には静的に割当てられない。

【0026】操作中、スケジューラー75は、各待ち行列について同等であるか又は所定の重みに応じた、r_i に等しい速度で各個別待ち行列iを処理する。速度 r_i で広い帯域幅を使用する特定の待ち行列（接続）（例えば、待ち行列30a）は、その他の待ち行列よりも長い待ち行列を有する傾向がある。全ての待ち行列接続が、それらの各帯域幅割当を完全に利用している場合、待ち行列30aはおそらくバッファオーバーフローを経験するであろう。

【0027】或る割当てメモリ（例えば、30c）が完全に利用されていない場合、本発明の公平キューイング（FQ）スキームは、待ち行列30bに到着するデータパケットのために、必要に応じて、十分に利用されていない待ち行列からバッファスペースを利用するか又は借りることができる。従って、バッファ待ち行列30aの予約割当て b_i を越えることができる。

【0028】第1の高速バッファ待ち行列30aのためのパケットを受信する第2のバッファが満杯になると、別の十分に利用されていないバッファ（例えば、待ち行列30i）は、待ち行列30aに宛てられる新たなデータパケットにバッファスペースを貸す。

【0029】同時に2つ以上の待ち行列がバッファオーバーフローを経験することもでき、そのため、十分に利用されていないバッファスペースから借りることもできる。従って、2つ以上の待ち行列がその予約割当て b_i を越えることもできる。本発明は、TCP接続ネットワークのフォワード及びリバース接続の両方に適用可能であり、また、データ及びACKトラヒックフローの両方に同等に適用可能である。

【0030】接続iが（b_i+1）個以上のバッファを必要とする場合、総占有率がB未満であるとすれば、使用可能なプールからスペースが割当てられる。

【0031】図3（a）は、本発明の接続毎の待ち行列スキームを実行するための一般的な流れ図である。ステップ201に示されるように、接続iについて宛てられたパケットが到着すると、最初のステップ203で、バッファBの現に残っている使用可能なスペースを有するカウンタをチェックし、新たに到着したパケットを確実に収容するための十分なメモリスペースがバッファ内に残存しているか否か決定する。

【0032】新たに到着したパケットを確実に収容するための十分なメモリスペースがバッファ内に残存している場合、ステップ208で処理が継続され、到着パケットが属する接続iを識別し、そして、ステップ211で、このパケットを、この接続に対応する待ち行列に格納する。このスキームに潜在的に含まれることは、到着パケットが適正に分類され、そして、待ち行列のうちの一つに割当てられることである。

【００３３】ステップ２０３において、新たに到着したパケットを確実に収容するための十分なメモリスペースがバッファ内に残存していないと決定される場合（例えば、現在の占有率q_iがb_i未満である待ち行列３０ｊがバッファを必要とする場合）、ステップ２２５において追出しスキームが実行され、到着パケットのために場所を空ける。特に、呼出されたＴＣＰプロトコルの実行に応じて、追出しが行われる待ち行列を選択するための２つの方法が使用できる。

【００３４】特に、図４（ｂ）に示されるように、第１の実施態様における追出しメカニズムは、ＬＱＦスキームであり、別の待ち行列から予約された最大量のメモリを借りる待ち行列、すなわち、（$q_i - b_i$）が全ての接続を通して最大であるような接続ｉを選択する。例えば、ステップ２５０に示されるように、バッファ内の各待ち行列の現在のバッファ割当てに関する決定が行われる。

【００３５】次いで、ステップ２６０で、各待ち行列の現在の待ち行列長さq_iを取得し、ステップ２７０で、各待ち行列の（$q_i - b_i$）差に関する計算を行う。最後に、ステップ２７５において、最大の（$q_i - b_i$）差を有する待ち行列を選択する。従って、その予約割当てb_iからの最大偏差は最長待ち行列であり、このため、図３（ａ）のステップ２２６で示されるように、到着パケットと先入れ相関関係を有する待ち行列からパケットがドロップされる。

【００３６】当業者ならば前記の最長待ち行列を最初に追出すスキーム（ＬＱＦスキーム）を実行するためのその他のアルゴリズムを案出することもできるので、本発明は図４（ｂ）に示された方法論に限定されない。例えば、最長遅延ファースト（“ＬＤＦ”）ドロッピングメカニズムは、割当て処理速度r_iが全て等しい場合、ＬＱＦスキームと同等に実行できる。なぜなら、待ち行列が同じ速度で処理される場合、遅延は各接続について同一だからである。同様に、処理速度が均一でない場合、待ち行列長さが同一であっても、遅延は異なる。従って、ＬＱＦスキームはＬＤＦの特例である。

【００３７】第２の最長待ち行列よりもかなり長い待ち行列を有する多数の接続を有するシステムで実行する場合、最長待ち行列ファースト（ＬＱＦ）は過度のバースト損失をもたらすことがある。すなわち、２個以上の非常に接近したパケットは、その割当てを越える待ち行列から連続的にドロップされる。例えば、ＴＣＰ－Ｒｅｎｏ型処理系はバースト損失の存在下で不良動作を行うことが知られているので、ＴＣＰ－Ｒｅｎｏアーキテクチャを実行する接続の性能は悪影響を受ける。従って、前記ＬＱＦにおけるバースト損失量を低下するために、スキームは、それぞれの割当てを越えるバッファからランダムに摘出するランダムジェネレータを使用するように変更されている。

【００３８】特に、第２の実施態様では、各バックログ接続（backlogged connection：予備接続）ｉは、$b_i = B/n$（ここで、ｎはバックログ接続の個数である）で示される名目バッファ割当てを有する。図５（ｃ）に示されるように、ステップ２５５で、各待ち行列のメモリ割当てb_iを取得する。次いで、ステップ２６５で、バックログ接続を例えば、２個のサブセットにグループ分けする。一方は、b_iよりも大きな占有率q_iを有するグループであり、他方は、$q_i \leq b_i$の関係を有するグループである。

【００３９】割当てより上（すなわち、$q_i > b_i$）の待ち行列群から、ステップ２８５で示されるように、１つの待ち行列がランダムに選択され、そして、ステップ２２６で示されるように、１つのパケットが先頭からドロップされる。

【００４０】異なる接続に関するバッファ占有率を平坦化し、そして、システムに過負荷をかける接続から最適に保護するための試みにおいて、L. Georgiadis, I. Cidon, R. Guerin, and A. Khamisy, “Optimal Buffer Sharing,” IEEE J. Select. Areas Commun., vol. 13, pp. 1229-1240, Sept. 1995に記載されるような開ループトラヒックについて実行されるスキームと同様な方法で、特定の追出し（pushout）スキームは最長待ち行列の先頭からパケットをドロップする。

【００４１】図６に示されるように、ハードウエアの観点から、カウンタ９０ａ，・・・，９０ｉは、それぞれの付随待ち行列の占有率q_iのトラックを保持するようにプログラムされたコントロールプロセッサ９２と共に、各待ち行列３０ａ，・・・，３０ｉに関連するように図示されている。従って、図４（ｂ）及び図５（ｃ）のステップ２６０及び２６５において、現在の待ち行列長さは、例えば、ポーリングにより、又は、図６のテーブル９９のようなレジスタテーブル（ここで、レジスタは最長待ち行列長さを示す）から位置指定することにより取得される。

【００４２】別のカウンタ９５は、総バッファ占有率Ｂのトラックを保持するために図示されている。従って、パケットが到着すると、プロセッサ９２は、カウンタ９５で使用可能な総メモリを決定するために、ステップ２０３（図３（ａ）参照）におけるチェックを行う。

【００４３】最長待ち行列を決定するために、プロセッサ９２はソート済み構造を実行することもできる。例えば、各キューに入れる、キューから外す又はドロップ操作の時点で、待ち行列の対応カウンタがそれに応じて増分又は減分された後、最長待ち行列構造を常に知っておくために、その待ち行列占有率値q_iを現在の最長待ち行列占有率値と比較するように、ソート済み構造を実行する。

【００４４】これらフロー毎のスキームの変法は、指数関数的に増大するサイズを有する一連のビンを使用する

ことにより効率的に実行することができる。待ち行列が変更される場合（例えば、キューに入れる、キューから外す又はドロップする場合）は常に、システムは最高占有ビンのトラックを保持しながら、適当なビン（例えば、現在のビンと同一のビン、一つ上のビン又は一つ下のビン）に移動させる。

【００４５】バッファが満杯の場合、最高占有ビンからの待ち行列が選択され、その最初のパケットがドロップされる。バッファがバイト単位で測定される場合、この操作は、可変パケットサイズのため、新たに到着したパケットを収容するために十分なスペースが解放されるまで、繰り返さなければならない。真のＬＱＤを行うために、最高占有ビンにおける待ち行列をソート済みリストとして維持するか、又はいつも最長待ち行列をサーチしなければならない。

【００４６】ＦＩＦＯ－ＲＥＤ及びＦＱ－ＲＥＤスキームと比較される本発明のバッファ管理スキームの改善された性能のシミュレーションについて説明する。図７は、このシミュレーションシステム１１９を示す概要図である。図７において、ソース１０１ａ，・・・，１０１ｆはルータ１２０への高速アクセスパスを有する。ルータ１２０は唯一のボトルネックであり、本発明の接続／フロー毎のバッファ管理スキームを実行するものである。

【００４７】比較は、ＴＣＰ－Ｔａｈｏｅ及びＴＣＰ－Ｒｅｎｏソース、バースト及び欲張りソース（ｇｒｅｅｄｙ ｓｏｕｒｃｅ）、リバースパス幅輳を有する片方向及び両方向トラヒックソース、及び広範囲に異なるラウンドトリップ時間の混合を用いて行われる。アクセスパス遅延は、様々なラウンドトリップ時間のモデルを作るために広範囲にわたって設定され、宛先はパケット毎にＡＣＫと仮定した。

【００４８】片方向トラヒックの場合、ＡＣＫは非輻輳パスを介して送信される。両方向トラヒックの場合、リターンパスはルータを経由し、そのため、キューイング（待合わせ）遅延が生じることがある。特に、ルータがＦＩＦＯスケジューリングを使用する場合、ＡＣＫ及びデータパケットは待ち行列内で混合される。公平キューイングの場合、ＡＣＫは別のフローとして処理される。非対称トラヒックの場合、リターンリンクの帯域幅は、宛先からルータまで低下される。そのため、相当なリバースパス輻輳とＡＣＫ損失が生じる。

【００４９】シミュレーションシステム１１９におけるＴＣＰ－Ｔａｈｏｅ及びＴＣＰ－Ｒｅｎｏの処理系は、４．３－Ｔａｈｏｅ　ＢＳＤ及び４．３－Ｒｅｎｏ　ＢＳＤのＴＣＰフローと輻輳制御挙動とをそれぞれモデル化する。ＲＥＤモデルは有向パケットであり、最小閾値としてバッファサイズの２５％を使用し、最大閾値としてバッファサイズの７５％を使用する。待ち行列重みは０．００２である。

【００５０】図８（ａ）及び（ｂ）は、従来技術によるＦＩＦＯ－ＲＥＤ及びＬＱＦ法と比較されるような、利用率及び公平性のそれぞれの観点から、擬似ＴＣＰネットワークで公平待ち行列ＬＱＤ及びＲＮＤドロップ法を実行した場合の、改善された性能を示す特性図である。

【００５１】特に、図８（ａ）及び（ｂ）は、図７の擬似ネットワークにおける１０Ｍｂｐｓ／１００ｋｂｐｓ容量（ＴＣＰ－Ｔａｈｏｅ及びＴＣＰ－Ｒｅｎｏの２０ｍｓ－１６０ｍｓラウンドトリップ時間）を有する非対称ボトルネックリンクによる２０個のＴＣＰ接続のためのバッファサイズ（で表したリンク速度）の関数として、スループット（図８（ａ））及び公平性係数（図８（ｂ））により評価される改善されたＴＣＰ性能を示す。

【００５２】図示されているように、ライン１３７ａ及び１３８ａでそれぞれ示されるＦＱ（公平キューイング）－ＲＥＤ及びＦＩＦＯ－ＲＥＤの両戦略は、ライン１３９及び１４０で示されるＦＱ－ＬＱＦ及びＦＱ－ＲＮＤドロップ法よりも非常に低いスループットを有する。なぜなら、再送パケットに対応するＡＣＫが、擬似非対称値が３（これは帯域幅非対称ではない）の場合の時間が６６％も喪失されるからである。これは、ＴＣＰサイクルの少なくとも６６％でタイムアウト（時間切れ）を生じ、スループットを著しく低下させる。

【００５３】フォワードパスにおける多数の損失及びフォワードパスにおける再送パケットの損失のために、その他のタイムアウトも起こる。一方、リバースパスにおける先頭からのドロップはこれらのタイムアウトを殆ど完全に除去する。タイムアウトは不経済なので、両方のＲＥＤスキームとも、ＦＩＦＯ－ＬＱＤを含むその他のスキームよりも著しく低いスループットを有する。

【００５４】更に、図８（ｂ）に示されるように、ＦＱ－ＲＮＤ及びＦＱ－ＬＱＤワークの両方とも非常に良好である。なぜなら、これらは、フロー毎の待ち行列の効果と、先頭からのドロップのタイムアウト除去効果とを併有するからである。ＦＱ－ＬＱＤは、再送パケットのドロップに対する組込みバイアスを有するという別の効果も有する。これは、ソースが最初の重複ＡＣＫの受信による損失を検出したときに、パケットの送信を中止するからである。

【００５５】第３の重複ＡＣＫが受信された後にだけ、再送パケットが送信される。介入間隔中、ソースがＴＣＰにより強制的に不活動化される場合、フローに対応する待ち行列は、少なくともその最小保証速度で排出される。従って、再送パケットが到着する場合、最長待ち行列であることは殆どない。従って、再送パケットのドロッピングに対する固有バイアスである。このバイアスは非対称ネットワークに限定されないが、低速リバースチャネル膨張のために、非対称要因、第１の重複ＡＣＫ受信と第３の重複ＡＣＫ受信との間の間隔により、バイア

スは非対称ネットワークにおいて高められる。

【００５６】再送パケットの損失は不経済なタイムアウトを起こすので、このバイアスは、図８（ｂ）においてライン１４１で示されるように、ＦＱ－ＬＱＤの性能を改善する。ライン１４２で示されるＦＱ－ＲＮＤも同様にこのバイアスを有するが、著しく低い。その理由は、ＦＱ－ＬＱＤに対するものと若干類似している。

【００５７】第１の重複ＡＣＫ受信と第３の重複ＡＣＫ受信との間の間隔中に、フローの待ち行列は、（ソースは不活動化されているので）少なくともその保証速度と等しい速度で流出し、待ち行列占有率はこのフローの予約パラメーター以下にまで落ちるであろう。この場合、再送パケットが到着すると、集合バッファが満杯であったとしても、再送パケットは失われない。これもの効果より、ＦＱ－ＬＱＤ及びＦＱ－ＲＮＤは最高の性能を有する。

【００５８】

【発明の効果】以上説明したように、本発明によれば、

長ラウンドトリップ時間を伴う接続に対するＴＣＰの固有の不公平性を軽減する、　異なるＴＣＰバージョンを用いる接続がボトルネックリンクを共用する場合、分離性を与える、　一層侵略的なトラヒックソース、不品行なユーザ又はリバースパス幅輳の場合のその他のＴＣＰ接続からの保護性を与える、　両方向トラヒックの存在下におけるＡＣＫ圧縮作用を軽減する、　ＡＣＫ損失（トラヒックのバーストを起こす）を経験するユーザに顕著な悪影響を及ぼすその他の接続から保護する、及び

全体的なリンク利用率を低下させることなく、“欲張り”接続を伴うボトルネックを共用する双方向接続に対する低待ち時間を与える、などのようなＴＣＰの種々の目標を達成することができる。

【図面の簡単な説明】

【図１】従来技術によるＴＣＰネットワーク接続を示すブロック図である。

【図２】多数のＴＣＰ接続のための共用バッファアーキテクチャのブロック図である。

【図３】（ａ）はバッファ割当て及びパケットドロッピングスキームを行うために履行される方法を示す流れ図であり、図３、図４、図５で１つの流れ図を形成する。

【図４】（ｂ）はＬＱＦパケットドロップ法を示す流れ図である。

【図５】（ｃ）はＲＮＤパケットドロップ法を示す流れ図である。

【図６】各バッファ待ち行列に関連するハードウエアを示すブロック図である。

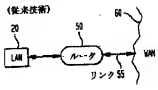
【図７】公平キューイングを処理するルータを有するＴＣＰ／ＩＰネットワークのシミュレーションを示すブロック図である。

【図８】（ａ）は図７の擬似ネットワークにおけるボトルネックリンクによる２０個のＴＣＰ接続のためのバッファサイズ（で表したリンク遅度）の関数として、スループットにより評価される改善されたＴＣＰ性能を示す特性図である。（ｂ）は図７の擬似ネットワークにおけるボトルネックリンクによる２０個のＴＣＰ接続のためのバッファサイズの関数として、公平性係数（全リンク容量に対する個々のスループットの割合の平均値に対する標準偏差の比）により評価される改善されたＴＣＰ性能を示す特性図である。

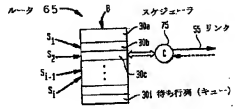
【符号の説明】

３０　待ち行列（キュー）
５５　ルータ
６５　ルータ
７５　スケジューラ
９２　コントロールプロセッサ
９５　カウンタ
９９　レジスタテーブル
１０１　ソース
１１９　シミュレーションシステム
１２０　ルータ

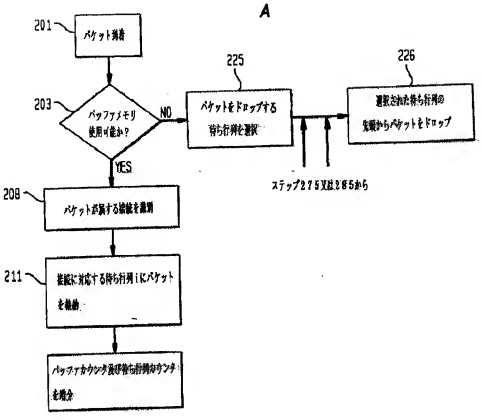
【図１】



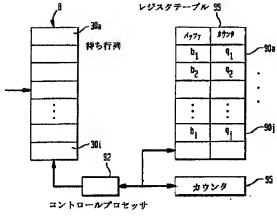
【図２】



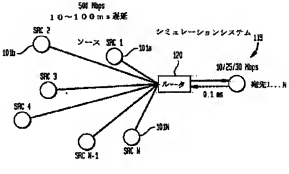
【図３】


A
201
パケット到着
203
バッファメモリ使用可能か？
NO
YES
225
パケットをドロップする待ち行列を選択
226
選択された待ち行列の先頭からパケットをドロップ
ステップ２７５又は２８５から
208
パケット格納する領域を識別
211
領域に対応する待ち行列にパケットを格納
バッファカウンタ及び待ち行列のカウンタを増分

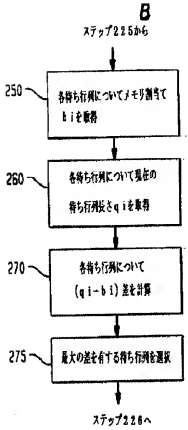
【図６】


8
30a
待ち行列
30i
レジスタテーブル 55
メップ
カウンタ
b_1
q_1
b_2
q_2
b_j
q_j
50a
50j
52
カウンタ
55
コントロールプロセッサ

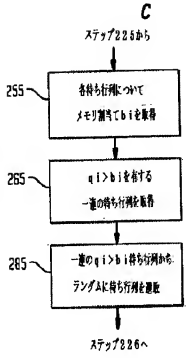
【図７】


500 Mbps
１０～１００ｍｓ遅延
シミュレーションシステム
110
SRC 2
10a
ソース
SRC 1
10b
120
SRC 3
SRC 4
ルータ
10/25/30 Mbps
0.1 ms
宛先1...N
SRC N-1
SRC N
10N

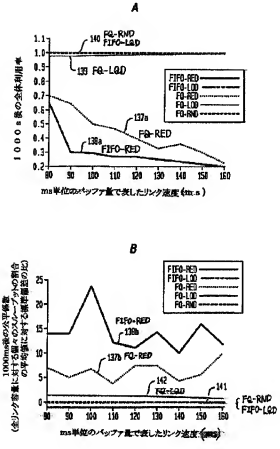
【図４】


B
ステップ２２５から
250
各待ち行列についてメモリ割当て
ｂｉを取得
260
各待ち行列について現在の
待ち行列長ｑｉを取得
270
各待ち行列について
(ｑｉ－ｂｉ) 差を計算
275
最大の差を有する待ち行列を選択
ステップ２２６へ

【図５】


C
ステップ２２５から
255
各待ち行列について
メモリ割当てｂｉを取得
265
ｑｉ＞ｂｉを有する
一連の待ち行列を取得
285
一連のｑｉ＞ｂｉ待ち行列から
ランダムに待ち行列を選択
ステップ２２６へ

【図８】



フロントページの続き

(71)出願人　596077259
600　Ｍｏｕｎｔａｉｎ　Ａｖｅｎｕｅ，Ｍｕｒｒａｙ　Ｈｉｌｌ，Ｎｅｗ　Ｊｅｒｓｅｙ　07974－0636U．Ｓ．Ａ．
(72)発明者　ティー．ヴィー．ラクシュマン
アメリカ合衆国，07724　ニュージャージー，イートンタウン，ヴィクトリア　ドライブ　118
(72)発明者　ディミトリオス　スティリアディス
アメリカ合衆国，07030　ニュージャージー，ホボケン，パーク　アヴェニュー　106
(72)発明者　バーナード　シュター
アメリカ合衆国，07747　ニュージャージー，アバーディーン，アイダホ　レイン　12